IT-Service & Engineering

IPT GMBH

Josef-Wirmer-Str. 104
60488 Frankfurt am Main

Tel: +49 (0)69-97640251
Fax: +49 (0)69-97640252
Hotline +49 (0)172-6711288
Email: **ipt@ipt-ffm.de**
URL: **www.ipt-ffm.de**

Excel - Tipp Nr. 1

Datumsberechnungen per Mausklick oder auf Tastendruck

Versionen: Excel 97, 2000, 20002/XP und 2003

Wenn Sie Termine in Excel verwalten gibt es immer wieder Situationen, in denen Sie Datumsangaben anpassen müssen. Der Liefertermin wird um einen Tag vorgezogen, der Abgabetermin um eine Woche verschoben oder Sie möchten die Terminliste aus dem letzten Jahr ganz einfach auf das neue Jahr anpassen. In diesem Tipp möchten wir Ihnen einige kleine VBA-Makros vorstellen, mit deren Hilfe Sie Datumsangaben in Zukunft ganz einfach neu setzen können. Erhöhen Sie das Datum der aktuellen Zelle beispielsweise mit jedem Tastendruck um einen Tage, eine Woche, einen Monat oder ein Jahr. Bei Bedarf lassen sich auch Bereiche markieren und gleich mehrere Termine in einem Arbeitsschritt neu ermitteln.

Die wichtigste VBA-Funktion zur Durchführung der gewünschten Berechnungen ist DateAdd mit der folgenden Syntax:

```
DateAdd(<Einheit>,<Anzahl>,<Ausgangsdatum>)
```

Dabei übergeben Sie zunächst mit <Einheit> eine Zeichenkette, die die gewünschte Berechnungsart angibt. Wir verwenden dabei die folgenden Einstellungen:

- "yyyy" = Jahr
- "q" = Quartal
- "m" = Monat
- "ww" = Woche
- "d" = Tag

Der Parameter <Anzahl> bestimmt dann die Anzahl der gewünschten Jahre, Quartale, Monate etc. wobei Sie mit positiven Werten vor und mit negativen Werten zurück rechnen können. <Ausgangsdatum> ist schließlich das Datum, auf dessen Basis die Berechnung durchgeführt werden soll.

Um nun per VBA, das Datum der aktuellen Zelle um einen ganzen Monat in die Zukunft zu verschieben, genügt die folgende Zeile:

```
ActiveCell.Value = DateAdd("m", 1, ActiveCell.Value)
```

Auf dieser Basis haben wir die folgende Prozedur entwickelt, die noch ein wenig mehr kann: Sie rufen sie einfach mit der gewünschten Einheit (Tag, Woche, Monat, Quartal oder Jahr werden repräsentiert durch 1, 2, 3 usw.) sowie der Zahl für die Berechnung auf und der Wert der aktuellen Zelle wird entsprechend neu gesetzt:

```
Sub procDatumsRechner(intEinheit As Integer, _
                      intAnzahl As Integer)
Dim varStartDatum As Variant
Dim strEinheit As String

  Select Case intEinheit

  Case 1
    strEinheit = "d" 'Tag
  Case 2
    strEinheit = "ww" ' Woche
  Case 3
    strEinheit = "m" 'Monat
  Case 4
    strEinheit = "q" 'Quartal
  Case 5
    strEinheit = "yyyy" 'Jahr
  Case Else
    MsgBox "Ungültiger Parameter für Einheit", _
    vbCritical, "Excel Weekly Datumsrechner"
    Exit Sub
  End Select

  varStartDatum = ActiveCell.Value

  If IsDate(varStartDatum) Then
     ActiveCell.Value = DateAdd(strEinheit, _
     intAnzahl, varStartDatum)
  End If

End Sub
```

Wichtig dabei ist, dass wir noch eine kleine Fehlerprüfung eingebaut haben. Der Inhalt der aktuellen Zelle wird zunächst der Variablen "varStartDatum" zugewiesen, um dann über "IsDate" feststellen zu können, ob die aktuelle Zelle überhaupt ein Datum enthält. Ist die Zelle leer oder enthält sie eine Zeichenkette bzw. eine beliebige Zahl, die kein Datum ist, passiert nichts. Andernfalls wird die entsprechende Berechnung mit DateAdd durchgeführt und das Ergebnis zurück geschrieben.

Auf Basis dieser Prozedur können Sie Excel nun ganz einfach so erweitern, dass das Datum der aktuellen Zelle auf Tastendruck um einen Tag hoch- oder heruntergezählt wird:

1. Geben Sie zunächst die oben gezeigte Prozedur "procDatumsrechner" ein.
2. Erfassen Sie in demselben Modul die folgenden Prozeduren, die den Datumsrechner jeweils mit Parametern zum Addieren bzw. Subtrahieren von einem Tag aufrufen:

```
Sub procPlusTag()
  procDatumsRechner 1, 1
End Sub

Sub procMinusTag()
  procDatumsRechner 1, -1
End Sub
```

- Wählen in Excel das Menü *Extras-Makro-Makros* an und markieren Sie den Eintrag "procPlusTag".

- Klicken Sie auf *Optionen* und geben Sie unter *Tastenkombination* den gewünschten Buchstaben für die die **Strg**-Tastenkombination ein, die Sie verwenden möchten. Zum Beispiel: **Strg+Q**.
- Nachdem Sie Ihre Einstellung mit *Ok* bestätigt haben, markieren Sie "procMinusTag", klicken auf *Optionen* und legen eine weitere Tastenkombination fest. Zum Beispiel: **Strg+W**.
- Nachdem Sie mit *Ok* bestätigt und den Makro-Dialog mit *Abbrechen* verlassen haben, können Sie in Zukunft jedes Datum in einer Tabelle mit den Tasten **Strg+Q** und **Strg+W** jeweils um einen Tag herauf- oder heruntersetzen.

Nach dem gleichen Prinzip können Sie nun kleine Makros und Tastenkombinationen für jede Datumsberechnung erstellen, die Sie benötigen. Zum Beispiel zum Ermitteln der nächsten oder vorhergehenden Woche:

```
Sub procPlusWoche()
  procDatumsRechner 2, 1
End Sub

Sub procMinusWoche()
  procDatumsRechner 2, -1
End Sub
```

Sollten Sie lieber mit der Maus anstelle der Tastatur arbeiten, können Sie die Prozeduren natürlich auch entsprechenden Symbolen zuordnen, indem Sie das Dialogfeld *Extras-Anpassen* (Registerkarte "Befehle") verwenden.

Möchten Sie schließlich nicht nur einzelne Zellen, sondern gleich einen kompletten Bereich mit Datumsberechnungen anpassen, können Sie Makros wie das folgende einsetzen:

```
Sub procDatumsbereichAnpassen()
Dim rngAktiveZelle As Range

  For Each rngAktiveZelle In Selection
    rngAktiveZelle.Activate
     procDatumsRechner 5, 1
  Next
End Sub
```

Wenn Sie mehrere Zellen markieren und dieses Makro starten, werden die ausgewählten Datumsangaben jeweils um ein Jahr hochgezählt. Eine Datumsliste aus dem letzten Jahr, in der Sie jeweils den 1. Tag eines Monats eingegeben haben, können Sie so beispielsweise in einem einzigen Arbeitsschritt auf das aktuelle Jahr anpassen.

Weitere Tipps finden Sie unter: www.ipt-ffm.de